

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
3 January 2003 (03.01.2003)

PCT

(10) International Publication Number
WO 03/001814 A1

(51) International Patent Classification⁷: **H04N 7/26,**
7/50, 7/36

(21) International Application Number: PCT/IB02/02362

(22) International Filing Date: 20 June 2002 (20.06.2002)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
01401697.6 26 June 2001 (26.06.2001) EP
01402666.0 16 October 2001 (16.10.2001) EP

(71) Applicant (for all designated States except US): KONIN-
KLIJKE PHILIPS ELECTRONICS N.V. [NL/NL];
Groenewoudseweg 1, NL-5621 BA Eindhoven (NL).

(72) Inventors; and

(75) Inventors/Applicants (for US only): CARRASCO,

Paul [FR/NL]; Internationaal Octrooibureau B.V., Prof.
Holstlaan 6, NL-5656 AA Eindhoven (NL). PES-
QUET-POPESCU, Beatrice [FR/NL]; Internationaal
Octrooibureau B.V., Prof. Holstlaan 6, NL-5656 AA
Eindhoven (NL).

(74) Agent: LANDOUSY, Christian; Internationaal Oc-
trooibureau B.V., Prof. Holstlaan 6, NL-5656 AA
Eindhoven (NL).

(81) Designated States (national): CN, JP, KR, US.

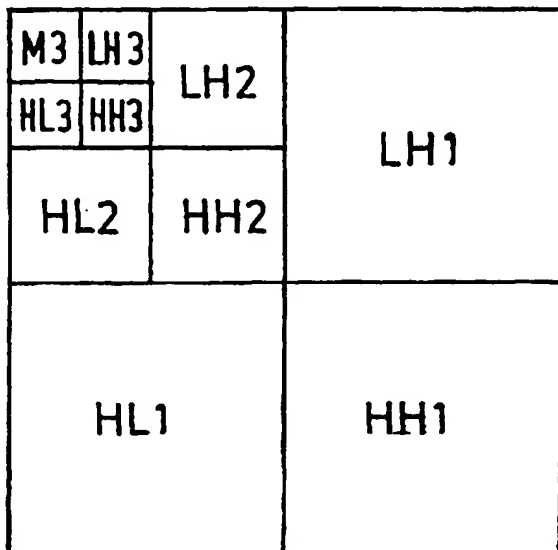
(84) Designated States (regional): European patent (AT, BE,
CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC,
NL, PT, SE, TR).

Published:

— with international search report

For two-letter codes and other abbreviations, refer to the "Guid-
ance Notes on Codes and Abbreviations" appearing at the begin-
ning of each regular issue of the PCT Gazette.

(54) Title: VIDEO CODING METHOD



(57) Abstract: The invention relates to a video coding method based on a subband coding scheme applied to a sequence of two-dimensional frames. Said method comprises a subband decomposition step of a current frame (F2), a motion prediction step, carried out with respect to a previous or reference frame (F1), and a coding step. According to the invention, the motion prediction step is based on a redundant decomposition of the reference frame, according to the following procedure : in the approximation subband, the motion is estimated by means of a full search operation applied to each pixel of the current frame, and for the other subbands, the motion is estimated by means of a pel-recursive motion estimation algorithm using for the prediction the reference redundant subband reconstructed from the subsampled subbands. Said pel-recursive motion estimation algorithm itself comprises an initialization sub-step, a determination and updating sub-step, and a decision sub-step.



WO 03/001814 A1

Video coding method

FIELD OF THE INVENTION

The present invention generally relates to the field of video compression and, more specifically, to a video coding method based on a subband coding scheme applied to a sequence of two-dimensional frames, said method comprising a subband decomposition step
5 of a current frame (F2), a motion prediction step, carried out with respect to a previous or reference frame (F1), and a coding step.

BACKGROUND OF THE INVENTION

In the multimedia domain, new interactive applications such as Internet video
10 streaming, video database browsing or multi-quality video services, are becoming widespread. This recent expansion of video services over heterogeneous networks (Internet, mobile nets, In Home Digital Networks) has raised new issues in terms of varying transport conditions (bandwidth, error rate...) as well as varying consumer demands and terminal decoding capabilities (CPU, display size, application...), and has led to investigate new
15 algorithms for video compression, in particular methods based on subband decompositions.

In the conventional video coding algorithms, and more particularly within the frame of the MPEG-4 standard, the motion estimation between successive frames of a processed video sequence is carried out by means of the so-called block-matching algorithm (BMA) : in BMA, a motion vector is assigned to a block of picture elements (pixels) in a
20 current frame (decomposed into blocks of fixed size) by searching a similar block within a defined area - or search window - of a reference frame, the best vector (that represents the shifting of the block) being found by evaluating candidates according to an error measure (the best corresponding block is the one that gives rise to a minimal error). The BMA unfortunately generates high frequency artefacts at the blocks edges (blocking effects) in the
25 motion compensated frames. When the BMA is used in coding schemes based on a wavelet decomposition, these artefacts reduce the coding efficiency by inducing high coefficients in the wavelet decomposition of the compensated frames.

SUMMARY OF THE INVENTION

The object of the invention is to propose a video coding method incorporating another type of motion estimation and compensation and allowing to improve the coding efficiency.

To this end, the invention relates to a method such as defined in the
 5 introductory paragraph of the description and which is moreover characterized in that said motion prediction step is based on a redundant decomposition of the reference frame, according to the following procedure :

(1) in the approximation subband of said decomposition, the motion is estimated by means of a full search operation involving said reference frame and applied to
 10 each picture element (pixel) of the current frame ;
 (2) for the other subbands, the motion is estimated by means of a pixel (or pel) recursive motion estimation algorithm using for the prediction the reference redundant subband reconstructed from the subsampled subbands yielded by said subband decomposition.

15 The proposed technical solution is based on the implementation of a pel-recursive motion estimation algorithm using wavelets. A motion estimation in the wavelet domain has to cope with the problem of translation invariance. The redundant decomposition of the reference frame is then used to predict the motion, allowing thus to take into account motion on the finest resolution grill. In the approximation subband, the motion is estimated
 20 by a full search algorithm applied to every pixel in the current frame. For the other subbands, a pel-recursive algorithm using for prediction the reference redundant subband is implemented. The initialization of this algorithm is made using a weighted value of the motion vectors of the spatial neighbours in the same subband and of the motion vector corresponding to the same position at the previous level of the decomposition. The algorithm
 25 allows to do a re-initialization at the positions failing to converge towards a good estimate. The scanning order in the subbands is also optimized in order to minimize the drift that can occur in a line-by-line scan.

More specifically, said pel-recursive motion estimation algorithm may comprise the following sub-steps:

30 (a) an initialization sub-step, the iterative algorithm being initialized with the mean value $d_0(m)$ of the motion vector in a causal neighbourhood S :

$$d_0(m) = \frac{1}{\text{Card}(S)} \sum_{j \in S} d(m - j)$$

the following notations being used :

m = pixel for which the motion has to be estimated

$d(m)$ = its motion vector ;

(b) an updating sub-step, for updating d_i according to the

relation :

- 5 $d_{i+1} = d_i + u_i$, where u_i , determined at each iteration i , is an update vector given by :

$$u_i = -\frac{E(m)}{\|\nabla A\|^2 + \lambda} \cdot \nabla A$$

where ∇A is the gradient of the image at the point $m-d_i$, $E(m) = B(m) - A(m-d_i)$ is the prediction error at the current iteration, and λ is a regularization parameter ;

(c) a decision sub-step, for carrying a break test and thus ending the procedure

- 10 if d_i is close enough to d ;

a typical neighbourhood of four pixels being preferably used.

In a particular implementation of the invention, the break test may be based on the estimation of the following ratio :

$$R = \frac{\|u_i\|^2}{\|d_i\|^2}$$

- 15 said ratio having to be greater than a given threshold ε .

Said break test may also include an additional condition put on a maximum number of iterations.

It may also be indicated that, according to an advantageous embodiment of the invention, the determination of said update vector u_i is based on a computation step including

- 20 the following minimizing operation : knowing that $B(m) = A(m-d_i) - \delta d^T \cdot \nabla A$ where

$\delta d^T \cdot \nabla A$ is the inner product of ∇A and vector $\delta d = d - d_i$, to minimize the square error J :

$$J = (B(m) - A(m-d_i) + \delta d^T \cdot \nabla A)^2 + \lambda \|\delta d\|^2$$

with respect to δd , the term $\lambda \|\delta d\|^2$ being a regularization term provided in view of a trade-

off between the smoothness of the resulting motion vector field, for large values of λ , and

- 25 the accuracy of the motion vectors, for small values of λ , the minimum being obtained for $\partial J / \partial \delta d = 0$ and leading to the update vector :

$$\delta d = u_i = -\frac{E(m)}{\|\nabla A\|^2 + \lambda} \cdot \nabla A$$

In another embodiment of the invention, the motion estimation algorithm may also comprise the following sub-steps :

(a) an initialization sub-step, the iterative algorithm being initialized with the mean value $d_{j,o}(m)$ of the motion vector in a causal neighbourhood S:

$$5 \quad d_{j,o}(m) = \frac{1}{\text{Card}(S)} \sum_{j \in S} d_j(m-k)$$

the following notations being used :

$k \in s$ (1 to 4)

m = pixel for which the motion has to be estimated

$d_{j,o}(m)$ = its motion vector ;

10 (b) an updating sub-step, for updating $d_{j,i+1}$ according to the relation :

$d_{j,i+1} = d_{j,i} + u_{j,i}$, where $u_{j,i}$ determined at each iteration i is an update vector given by :

$$u_{j,i} = \left(\sum_{s=0,1}^3 (\nabla A_{j,s} \nabla A_{j,s}^T) + \lambda I \right)^{-1} \left(- \sum_{s=0,1}^3 E_{j,s}(m) \nabla A_{j,s} \right)$$

where :

$$15 \quad \nabla A_{j,s} = \begin{pmatrix} \frac{\partial A_{j,s}(m-d_{j,i})}{\partial m} \\ \frac{\partial A_{j,s}(m-d_{j,i})}{\partial n} \end{pmatrix}$$

is the gradient of the image at point $m - d_{j,i}$

and $E_{j,s}(m) = B_{j,s}(m) - A_{j,s}(m - d_{j,i})$

(c) a decision sub-step, for carrying a break test and thus ending the procedure if $d_{j,i}$ is close enough to d_j .

20

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention will now be described, by way of example, with reference to the accompanying drawings in which:

Fig. 1 illustrates a subband decomposition of a video sequence, with motion 25 compensation ;

Figs. 2 and 3 respectively show a dyadic subband decomposition and a bank filter allowing to obtain said decomposition ;

Figs. 4 and 5 illustrate how the subsampling operation of such a dyadic subband decomposition does not preserve the translation invariance ;

Figs. 6 and 7 illustrate identities between representations, showing that redundant decompositions exist for each level of analysis;

Figs. 8 and 9 depict the so-called lifting scheme, respectively in the case of a signal decomposition and in the case of a signal reconstruction;

5 Fig. 10 to 12 show three levels of the classical wavelet decomposition obtained by applying the identities of Figs.6 and 7 to each bank filter of Fig.3;

Fig. 13 illustrates the choice of the samples (0 for even, 1 for odd) for a decomposition on three levels ;

10 Fig. 14 shows the 64 possible subsampling choices corresponding to the 64 possible decomposition bases;

Fig. 15 shows how interleaving subsampled subbands yields the redundant subband ;

Fig. 16 illustrates the redundant decomposition of a single frame, everyone of the ten redundant subbands having the same size as the original frame ;

15 Fig. 17 shows the classical, linear scanning order, and Fig.18 a causal scanning order allowing to obtain an improved causality in the initialization of the described pel-recursive algorithm ;

Fig. 19 shows a full search window in the redundant plane for the pixel (m,n).

20 DETAILED DESCRIPTION OF THE INVENTION

In order to obtain flexible video coding systems, able to cope with different requirements and capabilities, the scalability is the expected functionality. Progressive encoding techniques based on subband decompositions may be an answer, since they allow a fully progressive transmission. Particularly, wavelets have a high efficiency in progressively
25 encoding images in view of a scalable representation. They offer a natural multiscale representation for still images, that can be extended to video data by means of a 3D (or 2D+t) wavelet analysis including the temporal dimension within the decomposition (3D = three-dimensional ; 2D = two-dimensional ; t = time). The introduction of a motion compensation step in the 3D subband decomposition scheme leads to a spatio-temporal multiresolution
30 (hierarchical) representation of the video signal, as illustrated in Fig.1 showing a temporal subband decomposition of the video information, with motion compensation.

The illustrated 3D wavelet decomposition with motion compensation is applied to a group of frames (GOF), referenced F1 to F8. In this 3D subband decomposition scheme, each GOF of the input video is first motion-compensated (MC) in order to process

sequences with large motion, and then temporally filtered (TF) using Haar wavelets. The dotted arrows correspond to a high-pass temporal filtering, while the other ones correspond to a low-pass temporal filtering, and three stages of decomposition are shown (L and H = first stage ; LL and LH = second stage ; LLL and LLH = third stage).

5 However, when performing a pel-recursive motion estimation on the coefficients of each subband, the lack of translation invariance must be taken into account. The major problem with a dyadic subband decomposition (see Fig.2, showing such a decomposition, performed for three layers and leading to ten subbands LH1, HL1 and HH1 in the lowest layer, LH2, HL2 and HH2 in the middle layer, and M3, LH3, HL3 and HH3 in the highest layer ; and Fig.3, depicting a bank filter that allows to obtain this decomposition from an input signal $f[n]$, giving output signals $L1[n]$, $L2[n]$, $L3[n]$, $H1[n]$, $H2[n]$ and $H3[n]$) is indeed the fact that the subsampling operation (31, 32, 33, 34, 35, 36) applied after each filtering operation (21, 22, 23, 24, 25, 26) does not preserve the translation invariance (while the redundant wavelet transform of a signal is a time invariant representation, on the contrary the non redundant wavelet decomposition lacks this property). An example is given in Figs 4 and 5, where the samples in the high and low filtering branches, at the outputs of the low-pass and high-pass filters ($L(z)$ and $H(z)$) and the subsamplers ($\downarrow 2$), are shown respectively in the case of a first position of the edge of a moving object (original signal ORS) and in the case of a shifted input (shifted input SHF) corresponding to another position of said edge.

15 The obtained signal is shown just after each filter and each subsampler. Each subsampling operation has to keep either the even or the odd samples : a moving edge has resulted in a high frequency coefficient HFC in the first case (Fig.4, high filtering branch), while said edge is no longer detectable in the second one (Fig.5). Consequently, one cannot deduce motion from subsampled subband coefficients. The motion prediction must be performed from unsampled subbands.

20 For a 1D signal, a subsampling operation yields two possible choices : either odd or even samples can be taken. For a 2D signal (such as an image), this operation yields four possible decompositions. Therefore, if a signal frame is decomposed upon three levels, 64 possible decompositions, equivalent in terms of quantity of information, can be obtained. However, as previously mentioned, a motion estimation cannot be

25 performed on them, because they are not redundant. It is then observed from Fig.6 (illustrating the identity between the two representations $(L(z^2) + \downarrow 2)$ and $(\downarrow 2 + L(z))$ shown in that figure) that redundant decompositions exist for each level of analysis. The introduction of the equivalence of Fig.6 in the low filtering branch (21, 31, 22, 32, 23, 33) of

30 For a 1D signal, a subsampling operation yields two possible choices : either odd or even samples can be taken. For a 2D signal (such as an image), this operation yields four possible decompositions. Therefore, if a signal frame is decomposed upon three levels, 64 possible decompositions, equivalent in terms of quantity of information, can be obtained. However, as previously mentioned, a motion estimation cannot be performed on them, because they are not redundant. It is then observed from Fig.6 (illustrating the identity between the two representations $(L(z^2) + \downarrow 2)$ and $(\downarrow 2 + L(z))$ shown in that figure) that redundant decompositions exist for each level of analysis. The introduction of the equivalence of Fig.6 in the low filtering branch (21, 31, 22, 32, 23, 33) of

Fig.3 (repeated in the upper part of Fig.7) leads to the equivalent scheme of the lower part of said Fig.7 : now, before the final subsampling operation, a redundant subband whose size is the same as that of the original image is obtained, and a valid motion estimation can be performed on it (the redundant subbands RSB1, RSB2, RSB3 at the three levels 1, 2, 3 have been shown in Fig.7).

The wavelet decomposition itself is performed using the so-called lifting scheme, depicted for instance in Fig.8, in the case of the signal decomposition leading to outputs $c[n]$ and $d[n]$, and Fig.9, in the case of the signal reconstruction from said signals $c[n]$ and $d[n]$ (this implementation has been chosen because of the flexibility for the choice of the prediction filters P and update filters U , but it is obviously not the only one possible). For each filter pass, either odd or even samples ($x_1[n]$, $x_0[n]$ respectively) can be taken first, as it can be seen in Figs.8 and 9 :

- even samples first ? :

$$\left. \begin{array}{l} x_0[n] = f[2n] \\ x_1[n] = f[2n+1] \end{array} \right\} n \geq 0$$

- odd samples first ? :

$$\left. \begin{array}{l} x_0[n] = f[2n+1] \\ x_1[n] = f[2(n+1)] \end{array} \right\} n \geq 0$$

Figs.10 to 12 then show the "classical" wavelet decomposition obtained by applying the identities of Figs.6 and 7 to each filter bank of Fig.3 : one can see that stopping just before the decimation at each level (Fig.10, 11 or 12) yields the redundant subband.

The problem is that lifting wavelets inherently use decimation before filtering. In Fig.8, where the input signals for the lifting bank are $x_0[n] = f[2n]$, $x_1[n] = f[2n+1]$ or $x_0[n] = f[2n+1]$, $x_1[n] = f[2(n+1)]$, these inputs are decimated with respect to the original signal $f[n]$. Thus the decimation step cannot be skipped during the decomposition, and the redundant subbands can be obtained in another way, now described.

It has been seen, in relation with Figs 8 and 9, that for each level of decomposition and for each dimension, it is possible to choose even or odd samples first. If for instance the even choice is represented by 0 and the odd one by 1, a full decomposition on three levels is then characterized by six bits, as shown in the example of Fig.13 which can be seen as the binary decomposition of the number of the decomposition basis : 0,0,1,0,0,1 \rightarrow basis number 9. As a result, there are 64 possible subsampling choices corresponding to the 64 possible decomposition bases (as illustrated in Fig.14).

The critically subsampled subbands are then used to construct the redundant subband. The number of the decomposition basis yields the position where the subband will be placed on a rectangular grid. For example, if the bits describing x are 0 1 0 (3rd level, 2nd level, 1st level), they yield the number 2, which is the reconstruction offset showing where the first sample of the subband has to be placed in the redundant subband, following the x direction. The same operation is applied in the y direction, and the samples are then placed at intervals of length : $\text{interval} = 2^{\text{decomposition level}}$. Knowing the offset and the interval for each subsampled subband, it is possible to interleave them in order to reconstruct the redundant subband, as shown in Fig.15. This redundant subband has the same size as the original image.

At the first level of decomposition, each of the redundant subbands HL, LH and HH is obtained by interleaving four subsampled subbands. At level 2, they are obtained by interleaving 16 subsampled subbands, and at level 3, one has to interleave 64 subsampled subbands to obtain the redundant one (see Fig.14). Eventually, the redundant decomposition yields ten redundant subbands, as illustrated in Fig.16.

The pel-recursive motion estimation will be now described. In what follows, bold characters for vectors (\mathbf{d}), ∇ for the gradient, and $(.)^T$ for transposition will be used. The following algorithm is meant to compute the Optic Flow between two matrices A and B , which can be two successive frames of a sequence, or (as considered in the present case) two successive subbands. The aim is to estimate for each pixel $\mathbf{m} = (m, n)$, the motion vector $\mathbf{d}(\mathbf{m})$:

$$B(\mathbf{m}) = A(\mathbf{m} - \mathbf{d}) \quad (1)$$

To perform a recursive description of \mathbf{d} , it has to be assumed that \mathbf{d} is smoothly distributed over the image plane. Basically the method described here is : for each reference pixel in B , the gradient in A is computed at the estimated position $(\mathbf{m} - \mathbf{d})$ in order to find a pixel in A that is closer to the reference pixel. This yields a new position $\mathbf{m} - \mathbf{d}$. If it is the same position, the algorithm stops, else it iterates once more. Thus the algorithm converges and stops when it has reached the right position.

The iterative procedure is composed of the following steps, including the initialization, the computation of an update vector, the update of the motion vector, and a break test :

(1) The iterative procedure is initialized by the mean value of the motion vector in a causal neighbourhood S ,

$$\mathbf{d}_0(\mathbf{m}) = \frac{1}{\text{Card}(S)} \sum_{\mathbf{j} \in S} \mathbf{d}(\mathbf{m} - \mathbf{j}) \quad (2)$$

considering that a typical neighbourhood of four pixels can be used (the motion vectors are supposed to be zero out of the image boundaries) :

$$\begin{array}{ccccc} & * & j_1 & j_2 & j_3 & * \\ & * & j_0 & m & * & * \\ 5 & * & * & * & * & * \end{array}$$

(2) The second step is the determination, at each iteration i , of an update vector u_i :

$$u_i = - \frac{E(m)}{\|\nabla A\|^2 + \lambda} \nabla A \quad (3)$$

where :

$$\nabla A = \begin{pmatrix} \frac{\partial A(m-d_i)}{\partial m} \\ \frac{\partial A(m-d_i)}{\partial n} \end{pmatrix} \quad (4)$$

is the gradient of the image at point $m-d_i$,

$$E(m) = B(m) - A(m-d_i) \quad (5)$$

is the prediction error at the current iteration, and λ is a regularization parameter.

(3) In the third step, d_i is updated :

$$d_{i+1} = d_i + u_i \quad (6)$$

(4) The break test ends the procedure if d_i is close enough to d . This test can be for instance based on the estimation of the ratio :

$$R = \frac{\|u_i\|^2}{\|d_i\|^2} \quad (7)$$

which has to be superior to ϵ (ϵ can be turned to balance the vector determination precision and the computational load). A supplementary condition may be put on the maximum number of iterations.

The second step (update vector computation, described by the relation (3)) can be detailed. The first order approximation of the relation (1) reads :

$$B(m) = A(m-d_i) - \delta d^T \cdot \nabla A \quad (8)$$

where ∇A is the gradient from the relation (4) and $\delta d = d - d_i$

One has to minimize the square error (the term $\lambda \|\delta d\|^2$ will be explained later) :

$$\min_{\delta d} \left(\underbrace{(\beta(m) - A(m-d_i) + \delta d^T \cdot \nabla A)^2}_{J} + \lambda \|\delta d\|^2 \right) \quad (9)$$

The minimum is obtained when : $\frac{\partial J}{\partial \delta \mathbf{d}} = 0$, i.e. when :

$$(B(\mathbf{m}) - A(\mathbf{m} - \mathbf{d}_i) + \delta \mathbf{d}^T \cdot \nabla A) \cdot \nabla A + \lambda \delta \mathbf{d} = 0 \quad (10)$$

As $\delta \mathbf{d}^T \cdot \nabla A$ is the inner product of vectors $\delta \mathbf{d}$ and ∇A , one has : $\delta \mathbf{d}^T \cdot \nabla A = \nabla A^T \cdot \delta \mathbf{d}$ and the relation (10) becomes :

$$5 \quad (\nabla A \cdot \nabla A^T + \lambda I) \cdot \delta \mathbf{d} = -E(\mathbf{m}) \cdot \nabla A \quad (11)$$

The utility of λ can now be seen : the matrix $\nabla A \nabla A^T$ is not invertible (rank 1), but $\nabla A \nabla A^T + \lambda I$ can be inverted using the following lemma :

$$(M + uu^T)^{-1} = M^{-1} - \frac{M^{-1}uu^TM^{-1}}{1 + u^TM^{-1}u}$$

which yields here for $u = \nabla A$ and $M = \lambda I$:

$$10 \quad ((\nabla A \nabla A^T + \lambda I)^{-1} = \frac{\nabla A}{\lambda + \|\nabla A\|^2} \quad (12)$$

and finally leads to :

$$\delta \mathbf{d} = -\frac{E(\mathbf{m})}{\lambda + \|\nabla A\|^2} \nabla A \quad (13)$$

This is the update vector u_i used in the relations (3) and (6). This regularization provides a trade-off between the smoothness of the resulting motion vector field (for large λ) and the accuracy of the motion vectors (for small λ).

15 It has been seen, in the relation (11), that ∇A has to be computed. The gradient computation can be approximated by a discrete filtering of the image. To perform a discrete approximation of the gradient by linear filtering, the filters must satisfy some conditions that will be now reminded. Assuming that h is a FIR filter, and considering the rectangular sampling of a continuous field $F(x, y)$, one has : $F[n, m] = F(nL_1, nL_2)$. The gradient of the image following the x direction is :

$$\begin{aligned} L_1 \cdot \frac{\partial F}{\partial x}(nL_1, nL_2) &= h_{n,m} * F[n, m] \\ &= \sum_{(k,l) \in S} h_{k,l} \cdot F[n - k, m - l] \end{aligned} \quad (14),$$

S being considered as a close neighbourhood of 9 pixels (the center pixel m and the eight pixels around). On the other hand, at first order, one has :

$$F[n-k, m-l] \approx F[n, m] - kL_1 \cdot \frac{\partial F}{\partial x}(nL_1, nL_2) - lL_2 \cdot \frac{\partial F}{\partial y}(nL_1, nL_2) \quad (15)$$

and combining the relations (14) and (15) yields :

$$\begin{aligned} L_1 \cdot \frac{\partial F}{\partial x}(nL_1, nL_2) &= \sum_{(k,l) \in S} h_{k,l} F[n, m] \\ &\quad - L_1 \sum_{(k,l) \in S} kh_{k,l} L_1 \cdot \frac{\partial F}{\partial x}(nL_1, nL_2) \\ &\quad - L_2 \sum_{(k,l) \in S} lh_{k,l} L_1 \cdot \frac{\partial F}{\partial y}(nL_1, nL_2) \end{aligned} \quad (16)$$

Identifying the members of the relation (16) then yields three constraints for h :

$$\begin{aligned} 5 \quad &\sum_{(k,l) \in S} h_{k,l} = 0 \\ &\sum_{(k,l) \in S} kh_{k,l} = -1 \\ &\sum_{(k,l) \in S} lh_{k,l} = 0 \end{aligned}$$

The chosen filters, which satisfy these conditions, are the following :

$$\begin{aligned} 10 \quad &\text{(a) x-dimension gradient filter :} \\ &\begin{matrix} -1/8 & 0 & 1/8 \\ -1/4 & 0 & 1/4 \\ -1/8 & 0 & 1/8 \end{matrix} \\ &\text{(b) y-dimension gradient filter :} \\ &\begin{matrix} -1/8 & -1/4 & -1/8 \\ 0 & 0 & 0 \\ 1/8 & 1/4 & 1/8 \end{matrix} \\ 15 \end{aligned}$$

which yields a good approximation of the discrete gradient.

Different improvements can be proposed for the implementation of the algorithm described above, especially in terms of accuracy of motion vector and speed. The speed of the algorithm can be increased by making it more rapidly convergent. The initialization step is thus crucial for the convergence speed. A good initialization has to take into account as much as possible the previously computed motion vectors. However, the assumption about the smoothness of the distribution of the motion vectors over the image plane can be untrue (for certain types of motions, for instance when objects in the plane are moving in different directions) at the boundaries of the objects. The algorithm has to be able

to detect this event, and to correct the initialization by breaking the smoothness of the motion vector field. It is possible to introduce a break test at the end of the initialization step (and even after each update of the motion vector). One has to compute two test values:

$$E_0(m) = |B_{j,s}(m) - A_{j,s}(fm)|$$

$$5 \quad E_{j,s}(m) = |B_{j,s}(m) - A_{j,s}(fm - fd_i)|,$$

where E_0 is the error without motion ($d_i = 0$), and $E_{j,s}$ is the error when the computed motion vector is taken into account. The principle is that if $E_{j,s}$ is greater than E_0 , the motion vector has to be reinitialized to zero. However, too frequent reinitializations will prevent the algorithm from converging. When the prediction error is very small, this case could appear quite often. That is why a tolerance in this test is introduced, which is defined through inequality :

$$E_{j,s} > E_0 + \text{THR} \quad (17)$$

where THR is a threshold, to be determined according to the value of the subband coefficients.

15 Another improvement relates to the scanning order. The image being a table, it is scanned from the first item (i.e. $m = (0,0)$) to the last one ($m = x_{\max}, y_{\max}$), as shown in Fig.17. This initialization order, combined with the initialization neighbourhood, yields a bad motion estimation in the lower left corner of the image (since, for the first column of the image, the initialization neighbourhood is only reduced to the pixel above). The method is then not robust and leads to an error drift. The scanning order presented in Fig.18 yields better results. The image is scanned, the even lines from the left to the right, and the odd lines from the right to the left. This allows an improved causality in the initialization. Indeed, every pixel depends on all the previously scanned pixels. The error drift observed in the lower left corner of the image does not exist with this method. A better initialization makes possible for the algorithm to converge faster. The error drift was due to the fact that the number of iterations is limited : if the initialization is not good, the algorithm has not enough iterations to converge to the right motion vector. Given this new scanning order, the initialization neighbourhood has to be changed. For the first line, the neighbourhood is reduced to the previous pixel :

$$30 \quad * j_0 \ m \ *$$

For the even lines, the neighbourhood remains unchanged:

$$\begin{array}{ccccc} * & j_1 & j_2 & j_3 & * \\ * & j_0 & m & * & * \\ * & * & * & * & * \end{array}$$

A third improvement relates to the approximation subband SB $A_{j_{\max},0}$ of the image, which has, by definition, very few details. Thus a pel-recursive motion estimation is sometimes not accurate enough. Small errors in the compensated approximation yield bad results for the image reconstructed from the compensated subbands. Better results are
 5 obtained if a full search is performed for each pixel of $B_{j_{\max},0}$ in a window of $A_{j_{\max},0}$, but this method is not as fast as the pel-recursive algorithm. The size of the search window is :

$$(2.(2^{j_{\max}})+1) \text{ by } (2.(2^{j_{\max}}) + 1)$$

as shown in Fig.19. SB(3,0) being well estimated, it is possible to use its motion vector to initialize the other subband SB. If one calls $d_{\text{approximation}}$ the motion vector in SB(3,0) at pixel
 10 $m(m,n)$, and d_0 the initialization vector taking into account only spatial neighbours, a new initialization vector can be computed through :

$$D_0 = \alpha \cdot d_{\text{approximation}} + (1 - \alpha) \cdot d_0, \quad (18)$$

where α is a constant that allows to control the influence of $d_{\text{approximation}}$ in the initialization. If $\alpha = 0$, $d_{\text{approximation}}$ has no influence at all. The same principle can be used for the other
 15 subbands SB. It is then possible to use the previously computed motion vector, at a higher level of decomposition j , in order to initialize the algorithm. If one calls d_{j+1} the motion vector estimated at level $j+1$, the new initialization vector is : $D_0 = \alpha \cdot d_{j+1} + (1 - \alpha) \cdot d_0$.
 The generic algorithm described above can be applied between two subbands. However, if it is used, the correlation that exists between the subbands of a same level of decomposition is
 20 not taken into account : the computed motion vector should be the same for each pixel in the three subbands (or four when LL-subband is taken into account). It is then possible to decide to estimate, for a given level of decomposition $j \in \{1,2,... j_{\max}\}$, for each pixel $m = (m,n)$, the motion vector $d_j(m)$:

$$B_{j,s}(m) = A_{j,s}(m - d_j) \text{ for } s \in \{0,... 4\} \quad (19)$$

25 where :

- . $A_{0,0}$ represents the first image,
- . $B_{0,0}$ represents the second image,
- . $A_{j,s}$ represents the subband number s at the j -th decomposition level of the frame A :
- the subband 0 represents the approximation subband (SB),
- 30 the subband 1 represents the horizontal detail SB,
- the subband 2 represents the vertical detail SB,
- the subband 3 represents the diagonal detail SB.

The assumption of d being smoothly distributed over the subband planes has to be made just as previously.

As previously, the iterative procedure, carried out for each level of decomposition $j \in \{1, 2, \dots, j_{\max}\}$, comprises four steps including the initialization, the computation of an update vector, the update of the motion vector and a break test, but with some slight modifications :

- 5 The iterative procedure is initialized by the mean value of the motion vector in a causal neighbourhood S ,

$$d_{j,0}(m) = \frac{1}{\text{Card}(S)} \sum_{k \in S} d_j(m-k) \quad (20)$$

with for example a typical neighbourhood of 4 pixels (the motion vectors are supposed to be zero out of the image boundaries):

$$\begin{array}{ccccc} & * & k_1 & k_2 & k_3 & * \\ 10 & & & & & \\ & * & k_0 & m & * & * \\ & * & * & * & * & * \end{array}$$

2. Then an update vector $u_{j,i}$ is determined :

$$u_{j,i} = \left(\sum_{s=0,1}^3 (\nabla A_{j,s} \nabla A_{j,s}^T) + \lambda I \right)^{-1} \left(- \sum_{s=0,1}^3 E_{j,s}(m) \nabla A_{j,s} \right) \quad (21)$$

- 15 where :

$$\nabla A_{j,s} = \begin{pmatrix} \frac{\partial A_{j,s}(m-d_{j,i})}{\partial m} \\ \frac{\partial A_{j,s}(m-d_{j,i})}{\partial n} \end{pmatrix} \quad (22)$$

is the gradient of the image at point $m - d_{j,i}$ and :

$$20 \quad E_{j,s}(m) = B_{j,s}(m) - A_{j,s}(m - d_{j,i}) \quad (23)$$

s varying from 1 to 3 or from 0 to 3, depending on the number of subbands that have to be taken into account (4 for the deeper decomposition level, 3 for the others).

3. Then the motion vector $d_{j,i+1}$ is updated :

$$25 \quad d_{j,i+1} = d_{j,i} + u_{j,i} \quad (24)$$

4. The break test ends the procedure if $d_{j,i}$ is close enough to d_j , this test being for example the estimation of the ratio :

$$R_{j,i} = \frac{\|u_{j,i}\|^2}{\|d_{j,i}\|^2} \quad (25)$$

which has to be superior than ε (ε to be determined). An additional condition is put on the maximum number of iterations.

For the update vector computation, at the first order, the relation (1) becomes, for every s :

$$5 \quad B_{j,s}(m) = A_{j,s}(m - d_{j,s}) - \delta d_j^T \nabla A_{j,s} \quad (26)$$

where $\nabla A_{j,s}$ is the gradient from Eq. (4) and $\delta d_j = d_j - d_{jj}$.

As it is wanted to minimize the square error on all the subbands at the same time:

$$\min_{\delta d_j} \left(\underbrace{\sum_{s=0,1}^3 (B_{j,s}(m) - A_{j,s}(m - d_{j,i}) + \delta d_j^T \nabla A_{j,s})^2 + \lambda \|\delta d_j\|^2}_{J_i} \right) \quad (27)$$

the minimum is obtained when :

$$10 \quad \frac{\partial J_i}{\partial \delta d_j} = 0$$

i.e. :

$$\sum_{s=0,1}^3 ((B_{j,s}(m) - A_{j,s}(m - d_{j,i}) + \delta d_j^T \nabla A_{j,s}) \nabla A_{j,s} + \lambda \delta d_j) = 0 \quad (28)$$

15 The relation (28) then becomes :

$$\left(\sum_{s=0,1}^3 (\nabla A_{j,s} \nabla A_{j,s}^T) + \lambda I \right) \cdot \delta d_j = -E_{j,s}(m) \cdot \nabla A_{j,s} \quad (29)$$

Even without regularization, the matrix $\sum_{s=0,1}^3 \nabla A_{j,s} \nabla A_{j,s}^T + \lambda I$ can be invertible. It is a 2 by 2 matrix, and the inverted matrix can be computed explicitly. In this case, the regularization term λI is required only for the control of the motion vector field coherency.

In case this pel-recursive motion estimation algorithm should be applied to the redundant subband, the previous algorithm must be adapted. Let f be the scale factor characterizing the ratio between the size of the redundant subband and the size of the subsampled subband :

$$f = 2^j$$

where j is the decomposition level.

It is known that a redundant subband presents the lack translation invariance.

As a result, the pel-recursive motion estimation can be performed between a redundant subband (reference frame) and a subsampled subband (current frame). The motion vectors

- 5 are then (assuming that even samples have been retained for the subsampled decomposition of $B_{j,s}$) defined through :

$$B_{j,s}(m) = A_{j,s}(fm - fd_j) \quad (30)$$

The iterative procedure is then slightly different :

1. $d_0(m)$ is initialized through the relation (20).
 10 2. The update vector u_i remains unchanged, with the new error :

$$E_{j,s}(m) = B_{j,s}(m) - A_{j,s}(fm - f d_i) \quad (31)$$

and the new gradient :

$$\nabla A_{j,s} = \frac{1}{f} \left(\frac{\frac{\partial A_{j,s}(fm - fd_{j,i})}{\partial m}}{\frac{\partial A_{j,s}(fm - fd_{j,i})}{\partial n}} \right) \quad (32)$$

15

3. The update equation is unchanged.
 4. The break test remains unchanged.

CLAIMS:

1. A video coding method based on a subband coding scheme applied to a sequence of two-dimensional frames, said method comprising a subband decomposition step of a current frame (F2), a motion prediction step, carried out with respect to a previous or reference frame (F1), and a coding step, said method being moreover characterized in that
 5 said motion prediction step is based on a redundant decomposition of the reference frame, according to the following procedure :

(1) in the approximation subband of said decomposition, the motion is estimated by means of a full search operation involving said reference frame and applied to each picture element (pixel) of the current frame ;

- 10 (2) for the other subbands, the motion is estimated by means of a pixel (or pel) recursive motion estimation algorithm using for the prediction the reference redundant subband reconstructed from the subsampled subbands yielded by said subband decomposition.

- 15 2. A coding method according to claim 1, in which said pel-recursive motion estimation algorithm, intended to compute the optic flow between two matrices A and B representing two successive subbands i.e. to estimate for each pixel m its motion vector $d(m)$ in such a manner that $B(m) = A(m-d)$, itself comprises the following sub-steps :

- (a) an initialization sub-step, the iterative algorithm being initialized with the
 20 mean value $d_0(m)$ of the motion vector in a causal neighbourhood S :

$$d_0(m) = \frac{1}{\text{Card}(S)} \sum_{j \in S} d(m-j)$$

the following notations being used :

m = pixel for which the motion has to be estimated ;

$d(m)$ = its motion vector ;

- 25 (b) an updating sub-step, for updating d_i according to the relation :

$d_{i+1} = d_i + u_i$, where u_i , determined at each iteration i , is an update vector given by :

$$u_i = - \frac{E(m)}{\|\nabla A\|^2 + \lambda} \cdot \nabla A$$

where ∇A is the gradient of the image at the point $m-d_i$, $E(m) = B(m) - A(m-d_i)$ is the prediction error at the current iteration, and λ is a regularization parameter ;

- (c) a decision sub-step, for carrying a break test and thus ending the procedure
 5 if d_i is close enough to d .

3. A coding method according to claim 2, in which a typical neighbourhood of four pixels is used.

- 10 4. A coding method according to anyone of claims 2 and 3, in which said break test is based on the estimation of the following ratio :

$$R = \frac{\|u_i\|^2}{\|d_i\|^2}$$

said ratio having to be greater than a given threshold ε .

- 15 5. A coding method according to claim 4, in which said threshold ε is turned to balance the vector determination precision and the computational load.

6. A coding method according to anyone of claims 4 and 5, in which said break test includes an additional condition put on a maximum number of iterations.

20

7. A coding method according to anyone of claims 4 to 6, in which the determination of said update vector u_i is based on a computation step including the following minimizing operation : knowing that $B(m) = A(m-d_i) - \delta d^T \cdot \nabla A$ where $\delta d^T \cdot \nabla A$ is the inner product of ∇A and vector $\delta d = d - d_i$, to minimize the square error J :

25
$$J = ((B(m) - A(m-d_i) + \delta d^T \cdot \nabla A)^2 + \lambda \|\delta d\|^2$$

with respect to δd , the term $\lambda \|\delta d\|^2$ being a regularization term provided in view of a trade-off between the smoothness of the resulting motion vector field, for large values of λ , and the accuracy of the motion vectors, for small values of λ , the minimum being obtained for $\partial J / \partial \delta d = 0$ and leading to the update vector :

$$\delta d = u_i = - \frac{E(m)}{\|\nabla A\|^2 + \lambda} \cdot \nabla A$$

8. A coding method according to claim 7, in which the scanning order is carried out in the following way : even lines from the left to the right, odd lines from the right to the left, the initialization neighbourhood being then also modified for the first line, where it is reduced to the previous pixel, and for the odd lines, where it is inverted to preserve causality

9. A coding method according to claim 7, in which said motion estimation algorithm is re-initialized by testing a break condition with some tolerance, the prediction error when using the estimated motion vector being, for said test, compared to the prediction error with no motion estimation, plus a threshold.

10. A coding method according to claim 2, in which the initialization of the motion vector is done with a linear combination of the average over the neighbourhood S and the motion vector obtained from the subband with the same orientation, at the previous decomposition level.

11. A coding method according to claim 1, in which said pel-recursive motion estimation algorithm, intended to use the correlation that exists between the subbands of a same level of decomposition i.e. to estimate for a given level of decomposition $j \in \{1, 2, \dots, j_{\max}\}$ and for each pixel m its motion vector $d_j(m)$ in such a manner that $B_{j,s}(m) = A_{j,s}(m - d_j)$ for $s = \{0, \dots, 4\}$, the two matrices A and B representing two successive subbands, itself comprises the following sub-steps :

(a) an initialization sub-step, the iterative algorithm being initialized with the mean value $d_{j,0}(m)$ of the motion vector in a causal neighbourhood S :

$$d_{j,0}(m) = \frac{1}{\text{Card}(S)} \sum_{j \in S} d_j(m - k)$$

the following notations being used :

$k \in s (1 \text{ to } 4) ;$

$m = \text{pixel for which the motion has to be estimated} ;$

$d_{j,0}(m) = \text{its motion vector} ;$

(b) an updating sub-step, for updating $d_{j,i+1}$ according to the relation :

$d_{j,i+1} = d_{j,i} + u_{j,i}$, where $u_{j,i}$ determined at each iteration i is an update vector given by :

$$u_{j,i} = \left(\sum_{s=0,1}^3 (\nabla A_{j,s} \nabla A_{j,s}^T) + \lambda I \right)^{-1} \left(- \sum_{s=0,1}^3 E_{j,s}(m) \nabla A_{j,s} \right)$$

where :

$$\nabla A_{j,s} = \begin{pmatrix} \frac{\partial A_{j,s}(m-d_{j,i})}{\partial m} \\ \frac{\partial A_{j,s}(m-d_{j,i})}{\partial n} \end{pmatrix}$$

5

is the gradient of the image at point $m - d_{j,i}$ and :

$$E_{j,s}(m) = B_{j,s}(m) - A_{j,s}(m - d_{j,i})$$

(c) a decision sub-step, for carrying a break test and thus ending the procedure

if $d_{j,i}$ is close enough to d_j .

10

12. A coding method according to claim 11, in which a typical neighbourhood of four pixels is used.

13. A coding method according to anyone of claims 11 and 12, in which said

15 break test is based on the estimation of the following ratio :

$$R_{ij} = \frac{\|u_{j,i}\|^2}{\|d_{j,i}\|^2}$$

said ratio having to be greater than a given threshold ε .

14. A coding method according to claim 13, in which said break test includes an

20 additional condition put on a maximum number of iterations.

15. A coding method according to anyone of claims 13 and 14, in which the determination of said update vector u_i is based on a computation step including the following minimizing operation : knowing that $B_{j,s} = A_{j,s}(m-d_{j,i}) - \delta d_j^T \cdot \nabla A_{j,s}$ where

25 $\delta d_j^T \cdot \nabla A$ is the inner product of $\nabla A_{j,s}$ and the vector

$\delta d_j = d_j - d_{j,i}$, to minimize the square error :

$$\min_{\delta d_j} \left(\underbrace{\sum_{s=0,1}^3 (B_{j,s}(m) - A_{j,s}(m - d_{j,i}) + \delta_j^T \nabla A_{j,s})^2}_{J_i} + \lambda \|\delta d_j\|^2 \right) \quad (27)$$

with respect to δd_j , the term $\lambda \|\delta d_j\|^2$ being a regularization term provided in view of a trade-off between the smoothness of the resulting motion vector field, for large values of λ , and
 5 the accuracy of the motion vectors, for small values of λ , and the minimum being obtained for $\partial J_j / \partial \delta d_j = 0$.

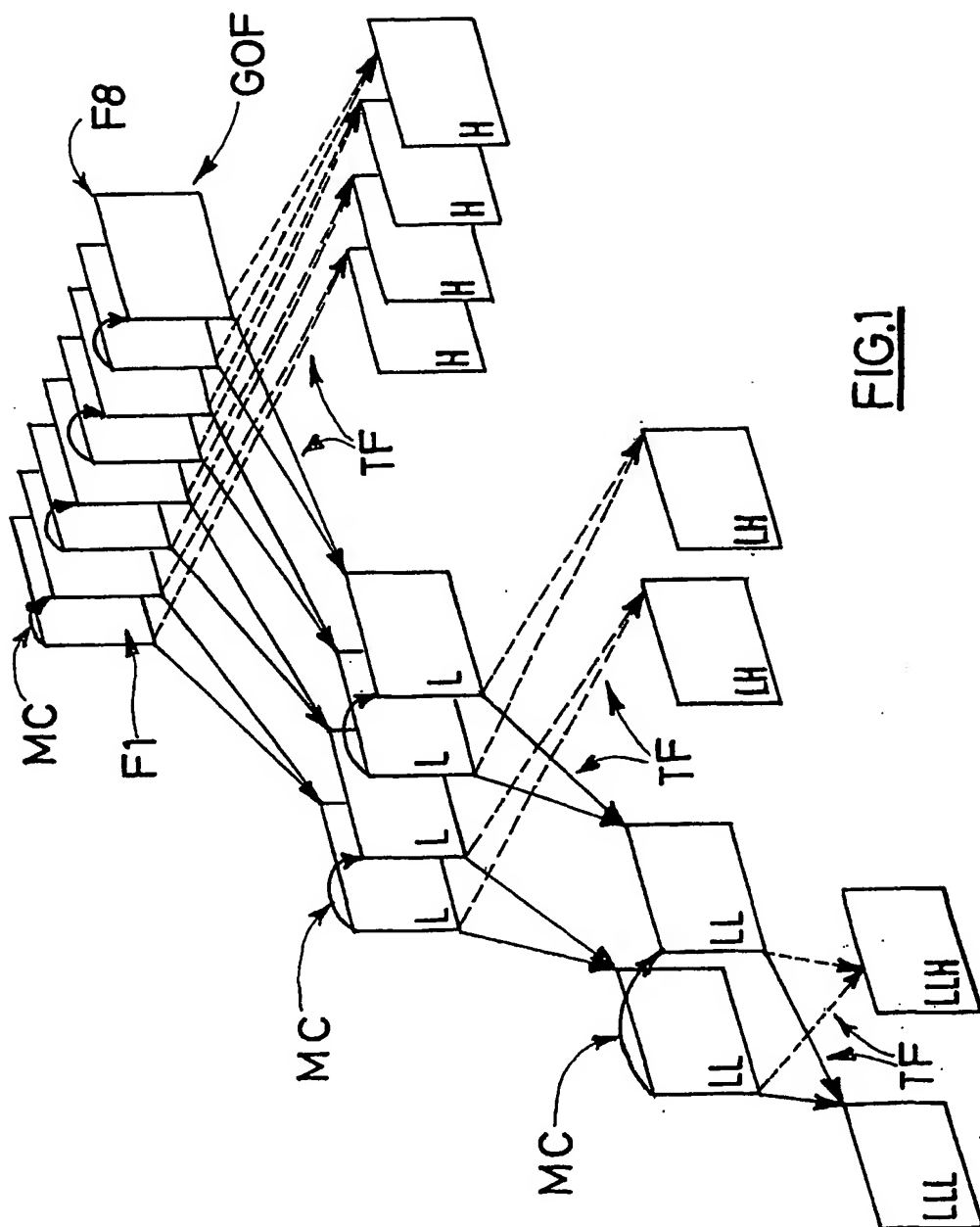
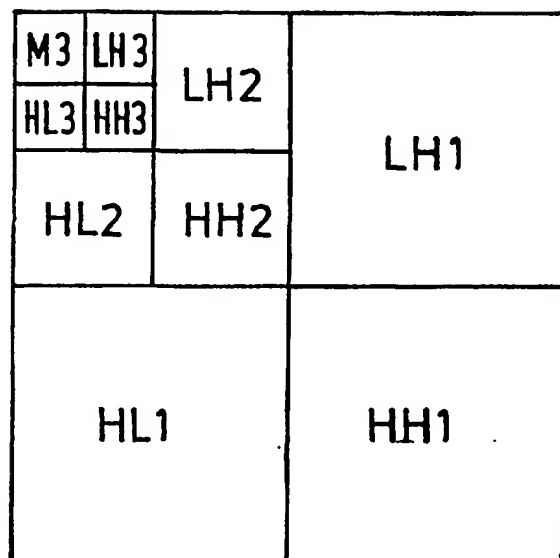
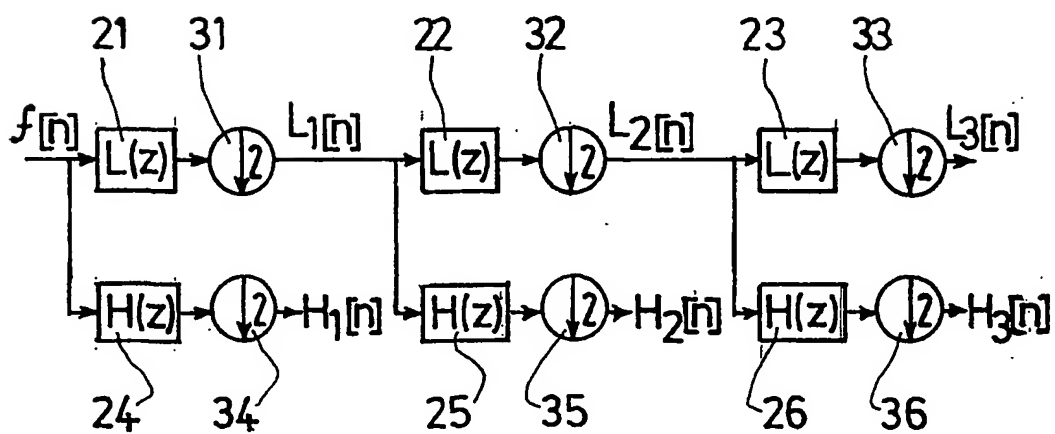


FIG. 1

2/10

FIG.2FIG.3

3/10

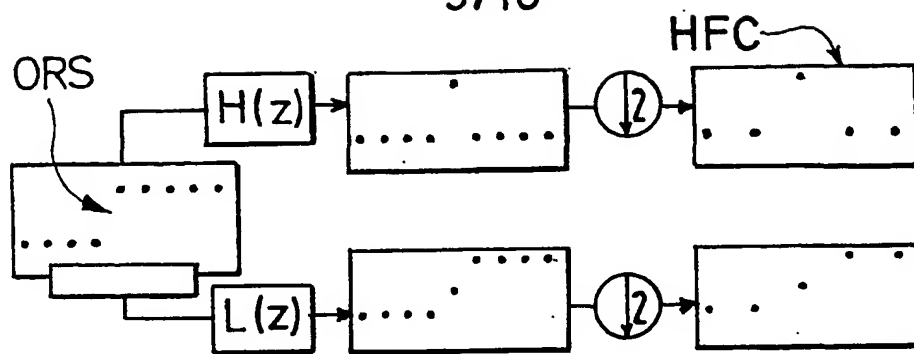


FIG. 4

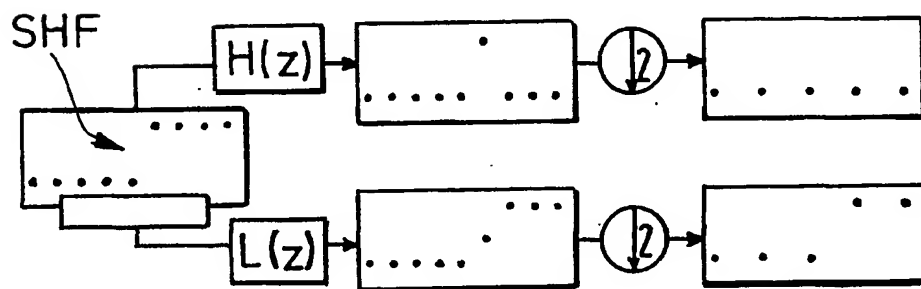
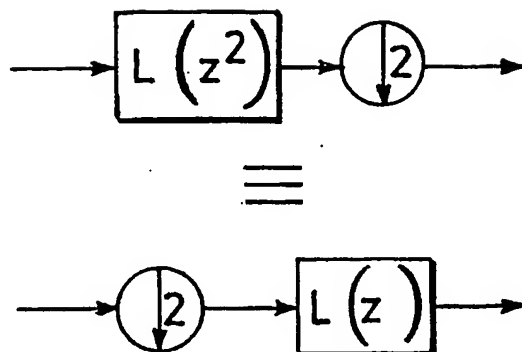
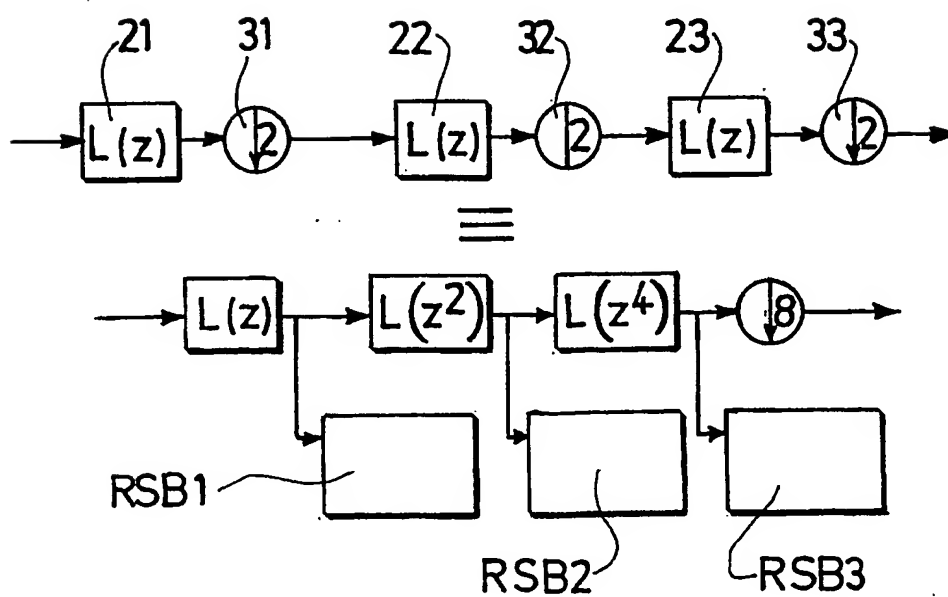
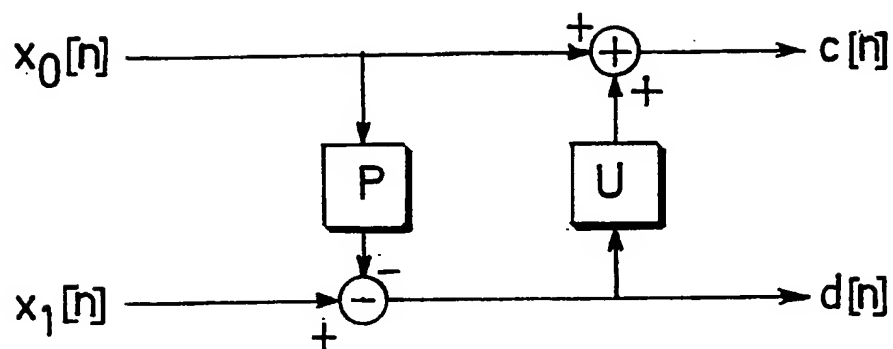
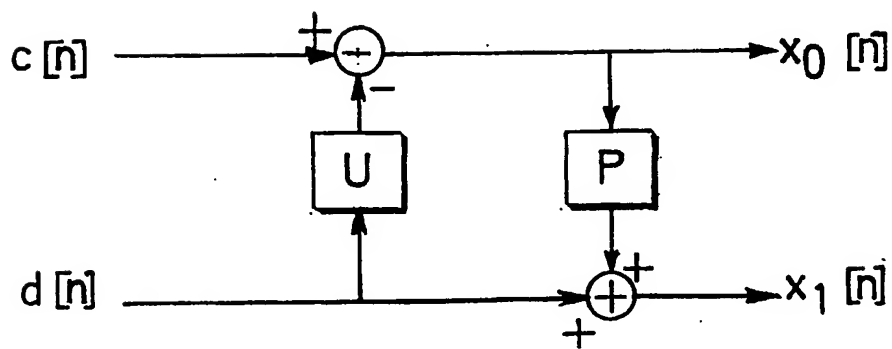


FIG. 5

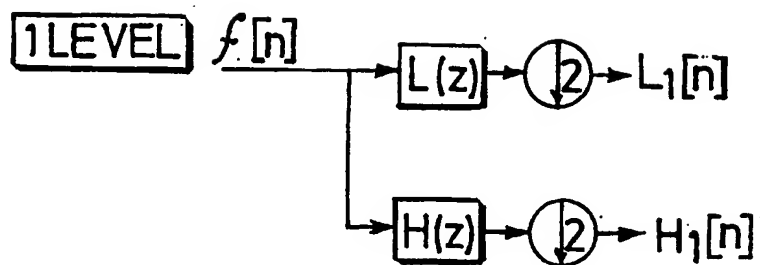
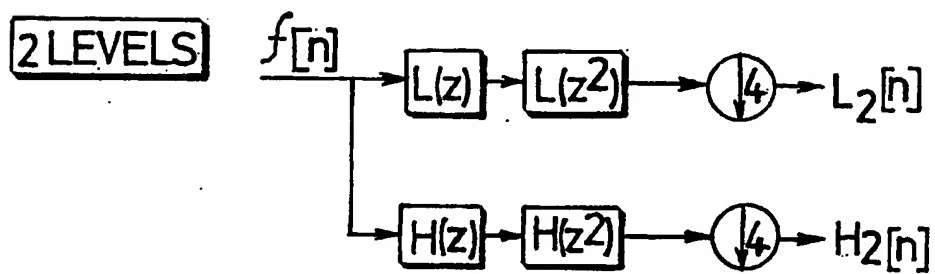
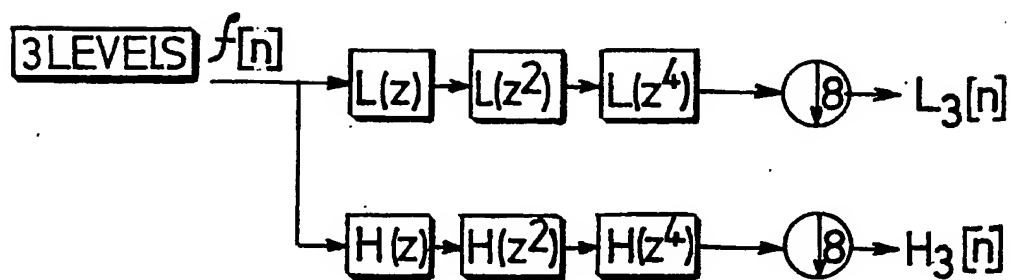
4/10

FIG.6FIG.7

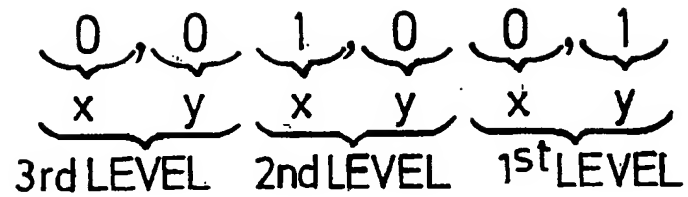
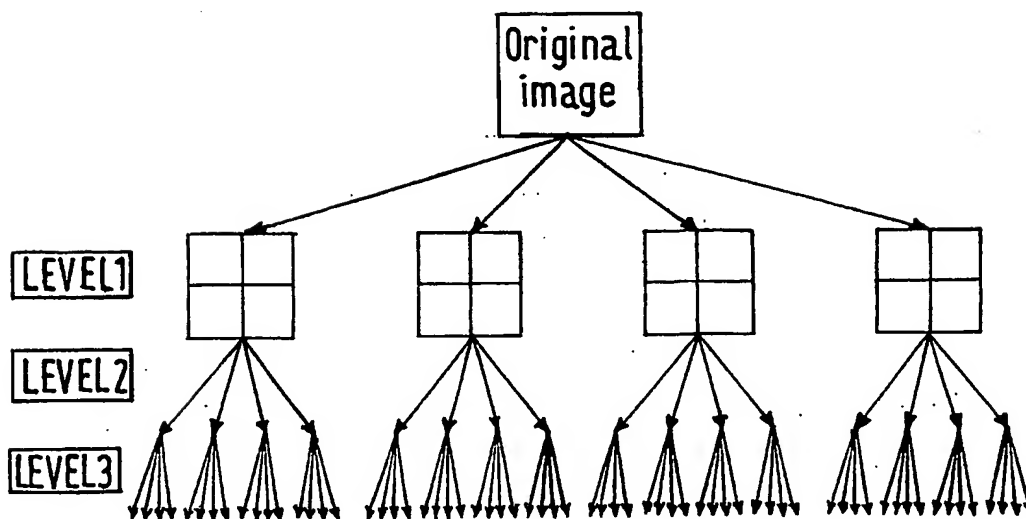
5/10

FIG. 8FIG. 9

6/10

FIG.10FIG.11FIG.12

7/10

FIG.13FIG.14

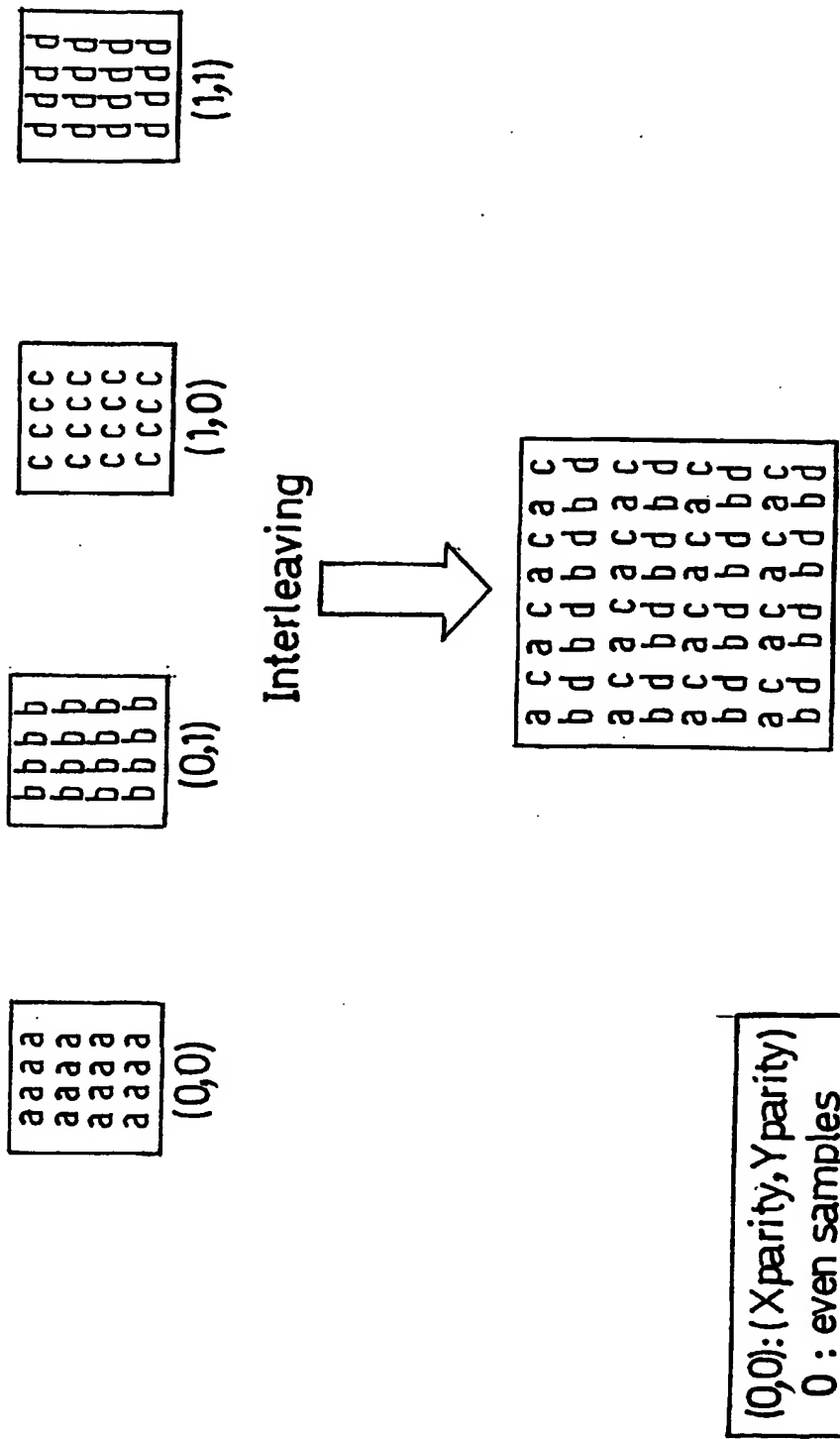


FIG.15

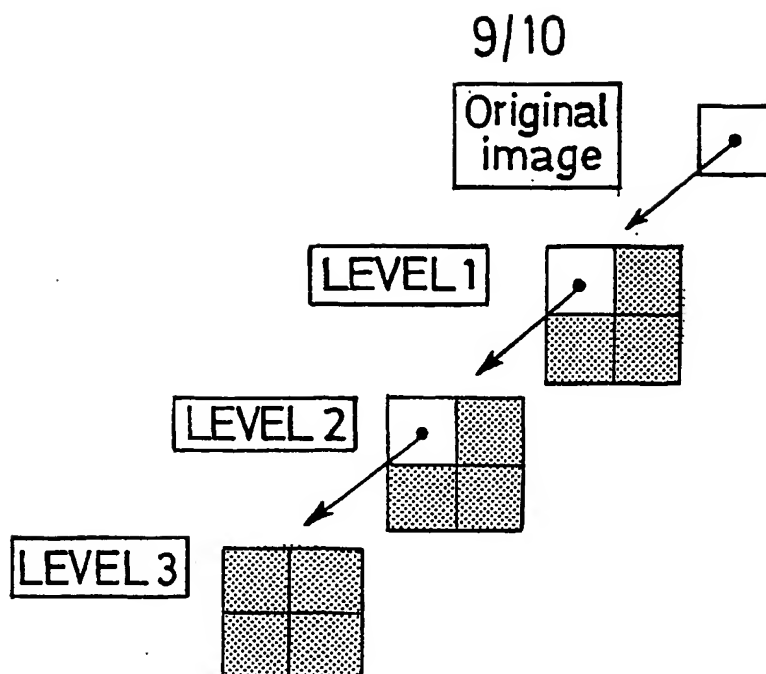


FIG.16

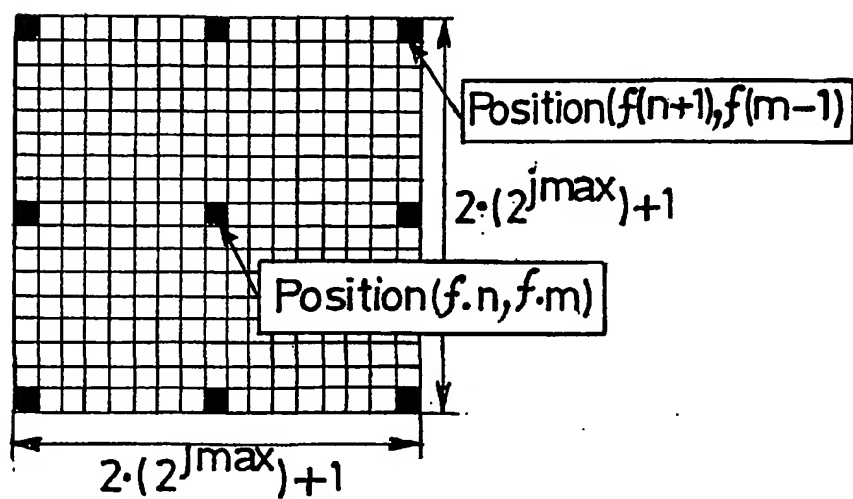


FIG.19

10/10

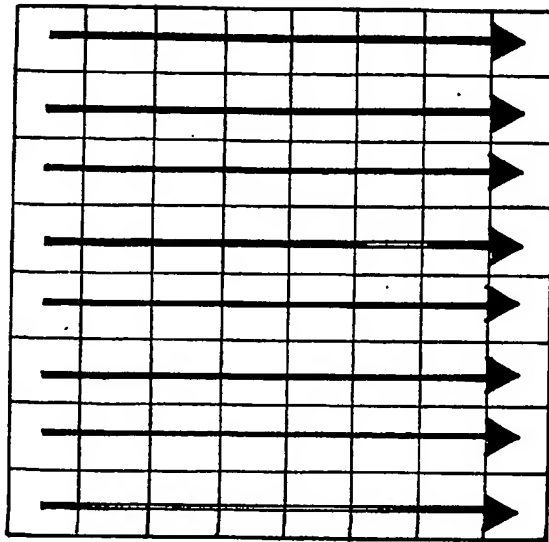


FIG.17

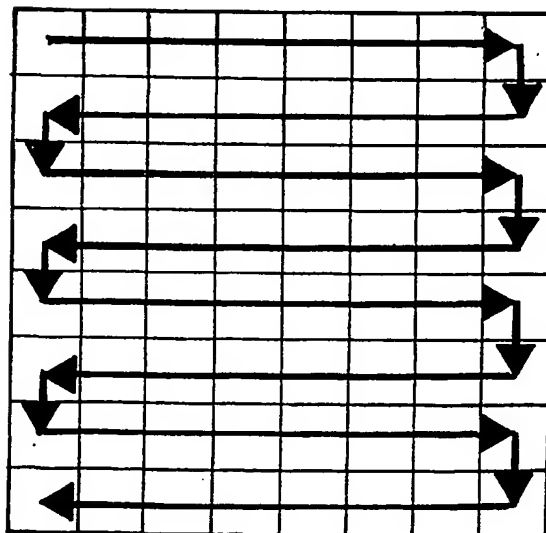


FIG.18

INTERNATIONAL SEARCH REPORT

In International Application No

PCT/IB 02/02362

A. CLASSIFICATION OF SUBJECT MATTER

IPC 7 H04N7/26 H04N7/50 H04N7/36

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 7 H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	<p>XUGUANG YANG ET AL: "Hierarchical backward motion compensation for wavelet video coding using optimized interpolation filters"</p> <p>IMAGE PROCESSING, 1997. PROCEEDINGS, INTERNATIONAL CONFERENCE ON SANTA BARBARA, CA, USA 26-29 OCT. 1997, LOS ALAMITOS, CA, USA, IEEE COMPUT. SOC, US, 26 October 1997 (1997-10-26), pages 85-88, XP010254114</p> <p>ISBN: 0-8186-8183-7</p> <p>page 87, paragraph 3</p> <p style="text-align: center;">— -/-</p>	1-15



Further documents are listed in the continuation of box C.



Patent family members are listed in annex.

* Special categories of cited documents:

A document defining the general state of the art which is not considered to be of particular relevance

E earlier document but published on or after the international filing date

L document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

O document referring to an oral disclosure, use, exhibition or other means

P document published prior to the international filing date but later than the priority date claimed

T later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

X document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

Y document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

Z document member of the same patent family

Date of the actual completion of the international search

19 September 2002

Date of mailing of the international search report

26/09/2002

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
 NL - 2280 HV Rijswijk
 Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
 Fax (+31-70) 340-3016

Authorized officer

Luckett, P

INTERNATIONAL SEARCH REPORT

In tional Application No
PCT/IB 02/02362

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	BOEROCZKY L ET AL: "MOTIONS ESTIMATION FOR WAVELET TRANSFORM CODING" JOURNAL ON COMMUNICATIONS, PROFINFO, BUDAPEST, HU, vol. 45, May 1994 (1994-05), pages 25-27, XPD01028196 ISSN: 0866-5583 page 26, paragraph 3.1 -----	1-15
A	US 5 253 058 A (GHARAVI HAMID) 12 October 1993 (1993-10-12) column 8, line 35 -column 8, line 66; figure 4 -----	1-15
A	US 4 969 040 A (GHARAVI HAMID) 6 November 1990 (1990-11-06) column 7, line 66 -column 8, line 14 -----	1-15
A	WO 00 13420 A (PIRIM PATRICK ;HOLDING B E V SA (LU)) 9 March 2000 (2000-03-09) claim 44 -----	1-15

INTERNATIONAL SEARCH REPORT
 ...formation on patent family members

Int'l Application No
PCT/IB 02/02362

Patent document cited in search report		Publication date	Patent family member(s)	Publication date
US 5253058	A	12-10-1993	WO 9320653 A1	14-10-1993
US 4969040	A	06-11-1990	NONE	
WO 0013420	A	09-03-2000	FR 2782876 A1	03-03-2000
			FR 2782878 A1	03-03-2000
			AU 5741799 A	21-03-2000
			WO 0013420 A2	09-03-2000
			EP 1110408 A2	27-06-2001